

DAFTAR PUSTAKA

- [1] J. W. Sheldon, "Recollections of the First Software Company," *IEEE Ann. Hist. Comput.*, vol. 16, no. 2, pp. 65–71, 1994, doi: 10.1109/85.279238.
- [2] N. Gupta *et al.*, "Forecast: Enterprise Application Software, Worldwide, 2016-2022, 1Q18 Update," *Research*, 2018. <https://www.gartner.com/doc/3869907/forecast-enterprise-application-software-worldwide> (accessed Apr. 23, 2019).
- [3] P. Pospieszny, B. Czarnacka-Chrobot, and A. Kobylinski, "An effective approach for software project effort and duration estimation with machine learning algorithms," *J. Syst. Softw.*, vol. 137, pp. 184–196, 2018, doi: 10.1016/j.jss.2017.11.066.
- [4] S. Grimstad, M. Jørgensen, and K. Moløkken-Østfold, "Software effort estimation terminology: The tower of Babel," *Inf. Softw. Technol.*, vol. 48, no. 4, pp. 302–310, 2006, doi: 10.1016/j.infsof.2005.04.004.
- [5] M. Bloch, S. Blumberg, and J. Laartz, "Delivering large-scale IT projects on time, on budget, and on value," *McKinsey Digital*, 2012. <https://mck.co/2yXnefx> (accessed May 23, 2019).
- [6] A. Ardiansyah, M. M. Mardhia, and S. Handayaningsih, "Analogy-based model for software project effort estimation," *Int. J. Adv. Intell. Informatics*, vol. 4, no. 3, p. 251, Nov. 2018, doi: 10.26555/ijain.v4i3.266.
- [7] M. Azzeh, A. B. Nassif, and S. Banitaan, "Comparative analysis of soft computing techniques for predicting software effort based use case points," *IET Softw.*, vol. 12, no. 1, pp. 19–29, 2017, doi: 10.1049/iet-sen.2016.0322.
- [8] A. W. M. M. Parvez, "Efficiency factor and risk factor based use case point test effort estimation model compatible with agile software development," in *2013 International Conference on Information Technology and Electrical Engineering (ICITEE)*, Oct. 2013, pp. 113–118, doi: 10.1109/ICITEED.2013.6676222.
- [9] M. Jorgensen and M. Shepperd, "A Systematic Review of Software Development Cost Estimation Studies," *IEEE Trans. Softw. Eng.*, vol. 33, no. 1, pp. 33–53, 2007.
- [10] M. Jorgensen and M. Shepperd, "A Systematic Review of Software Development Cost Estimation Studies," *IEEE Trans. Softw. Eng.*, vol. 33, no. 1, pp. 33–53, Jan. 2007, doi: 10.1109/TSE.2007.256943.
- [11] A. J. Albrecht, "Measuring Application Development Productivity," in *Proceedings of the Joint SHARE, GUIDE, and IBM Application Development Symposium*, 1979, pp. 83–92.
- [12] S. Di Martino, F. Ferrucci, C. Gravino, and F. Sarro, "Web Effort Estimation: Function Point Analysis vs. COSMIC," *Inf. Softw. Technol.*, vol. 72, pp. 90–109, 2016, doi: 10.1016/j.infsof.2015.12.001.
- [13] S. D. Conte, *Software Engineering Metrics and Models*. Benjamin-Cummings Pub Co, 1986.

- [14] S. Shukla and S. Kumar, "Towards non-linear regression-based prediction of use case point (UCP) metric," *Appl. Intell.*, Aug. 2022, doi: 10.1007/s10489-022-04002-4.
- [15] R. K. Clemmons, "Project Estimation With Use Case Points," *J. Def. Softw. Eng.*, 2006.
- [16] G. Schneider and J. P. Winters, *Applying Use Cases: A Practical Guide*, 2nd ed. Addison Wesley, 2001.
- [17] G. Karner, "Resource Estimation for Objectory Projects." 1993.
- [18] C. J. Neill and P. A. Laplante, "Requirements engineering: The state of the practice," *IEEE Softw.*, vol. 20, no. 6, pp. 40–45, Nov. 2003, doi: 10.1109/MS.2003.1241365.
- [19] M. Azzeh, A. Bou, and I. Basem, "Science of Computer Programming Predicting software effort from use case points : A systematic review," *Sci. Comput. Program.*, vol. 204, p. 102596, 2021, doi: 10.1016/j.scico.2020.102596.
- [20] A. B. Nassif, L. F. Capretz, and D. Ho, "Enhancing Use Case Points Estimation Method Using Soft Computing Techniques," *J. Glob. Res. Comput. Sci.*, vol. 1, no. 4, pp. 12–21, Dec. 2010, [Online]. Available: <http://arxiv.org/abs/1612.01078>.
- [21] M. Hariyanto and R. S. Wahono, "Estimasi Proyek Pengembangan Perangkat Lunak Dengan Fuzzy Use Case Points," *J. Softw. Eng.*, vol. 1, no. 1, pp. 54–63, 2015.
- [22] F. Wang, X. Yang, X. Zhu, and L. Chen, "Extended Use Case Points Method for Software Cost Estimation," in *2009 International Conference on Computational Intelligence and Software Engineering*, Dec. 2009, pp. 1–5, doi: 10.1109/CISE.2009.5364706.
- [23] Y. Xie, J. Guo, and A. Shen, "Use Case Points Method of Software Size Measurement Based on Fuzzy Inference," in *Proceedings of the 4th International Conference on Computer Engineering and Networks*, vol. 355, W. E. Wong, Ed. Cham: Springer International Publishing, 2015, pp. 11–18.
- [24] E.-G. Talbi, *Metaheuristics: From Design to Implementation*. Canada: Wiley Publishing, 2009.
- [25] Suyanto, *Algoritma Optimasi: Deterministik atau Probabilistik*, II. Bandung: Graha Ilmu, 2014.
- [26] A. M. Al-Zoubi, M. A. Hassonah, A. A. Heidari, H. Faris, M. Mafarja, and I. Aljarah, "Evolutionary competitive swarm exploring optimal support vector machines and feature weighting," *Soft Comput.*, vol. 25, no. 4, pp. 3335–3352, Feb. 2021, doi: 10.1007/s00500-020-05439-w.
- [27] J. Zhou *et al.*, "Optimization of support vector machine through the use of metaheuristic algorithms in forecasting TBM advance rate," *Eng. Appl. Artif. Intell.*, vol. 97, Jan. 2021, doi: 10.1016/j.engappai.2020.104015.
- [28] S. Anupam and A. K. Kar, "Phishing website detection using support vector machines and nature-inspired optimization algorithms," *Telecommun. Syst.*, vol. 76, no. 1, pp. 17–32, Jan. 2021, doi: 10.1007/s11235-020-00739-w.
- [29] E. H. Houssein, M. E. Hosney, D. Oliva, W. M. Mohamed, and M. Hassaballah, "A novel

- hybrid Harris hawks optimization and support vector machines for drug design and discovery,” *Comput. Chem. Eng.*, vol. 133, p. 106656, Feb. 2020, doi: 10.1016/j.compchemeng.2019.106656.
- [30] A. Tharwat and T. Gabel, “Parameters optimization of support vector machines for imbalanced data using social ski driver algorithm,” *Neural Comput. Appl.*, vol. 32, no. 11, pp. 6925–6938, Jun. 2020, doi: 10.1007/s00521-019-04159-z.
 - [31] M. Ghasemi, K. Bagherifard, H. Parvin, S. Nejatian, and K.-H. Pho, “Multi-objective whale optimization algorithm and multi-objective grey wolf optimizer for solving next release problem with developing fairness and uncertainty quality indicators,” *Appl. Intell.*, vol. 51, no. 8, pp. 5358–5387, Aug. 2021, doi: 10.1007/s10489-020-02018-2.
 - [32] M. Á. Domínguez-Ríos, F. Chicano, E. Alba, I. del Águila, and J. del Sagrado, “Efficient anytime algorithms to solve the bi-objective Next Release Problem,” *J. Syst. Softw.*, vol. 156, pp. 217–231, Oct. 2019, doi: 10.1016/j.jss.2019.06.097.
 - [33] C. Manjula and L. Florence, “Deep neural network based hybrid approach for software defect prediction using software metrics,” *Cluster Comput.*, vol. 22, no. S4, pp. 9847–9863, Jul. 2019, doi: 10.1007/s10586-018-1696-z.
 - [34] M. Anbu and G. S. Anandha Mala, “Feature selection using firefly algorithm in software defect prediction,” *Cluster Comput.*, vol. 22, no. S5, pp. 10925–10934, Sep. 2019, doi: 10.1007/s10586-017-1235-3.
 - [35] R. S. Wahono, “A Systematic Literature Review of Software Defect Prediction: Research Trends, Datasets, Methods and Frameworks,” *J. Softw. Eng.*, vol. 1, no. 1, pp. 1–16, 2015.
 - [36] M. Nevendra and P. Singh, “Empirical investigation of hyperparameter optimization for software defect count prediction,” *Expert Syst. Appl.*, vol. 191, no. November 2021, p. 116217, Apr. 2022, doi: 10.1016/j.eswa.2021.116217.
 - [37] A. Bajaj and O. P. Sangwan, “A Systematic Literature Review of Test Case Prioritization Using Genetic Algorithms,” *IEEE Access*, vol. 7, pp. 126355–126375, 2019, doi: 10.1109/ACCESS.2019.2938260.
 - [38] M. Khatibsyarbini, M. A. Isa, D. N. A. Jawawi, H. N. A. Hamed, and M. D. Mohamed Suffian, “Test Case Prioritization Using Firefly Algorithm for Software Testing,” *IEEE Access*, vol. 7, pp. 132360–132373, 2019, doi: 10.1109/ACCESS.2019.2940620.
 - [39] D. Wu, J. Li, and C. Bao, “Case-based reasoning with optimized weight derived by particle swarm optimization for software effort estimation,” *Soft Comput.*, vol. 22, no. 16, pp. 5299–5310, Aug. 2018, doi: 10.1007/s00500-017-2985-9.
 - [40] K. Langsari and R. Sarno, “Optimizing effort and time parameters of cocomo ii estimation using fuzzy multi-objective PSO,” *Int. Conf. Electr. Eng. Comput. Sci. Informatics*, vol. 4, no. September, pp. 466–471, 2017, doi: 10.11591/eecsi.4.1047.
 - [41] N. A. Zakaria, A. R. Ismail, N. Z. Abidin, N. H. M. Khalid, and A. Y. Ali, “Optimized COCOMO parameters using hybrid particle swarm optimization,” *Int. J. Adv. Intell. Informatics*, vol. 7, no. 2, p. 177, Apr. 2021, doi: 10.26555/ijain.v7i2.583.

- [42] F. Ferrucci, C. Gravino, R. Oliveto, and F. Sarro, "Genetic Programming for Effort Estimation: An Analysis of the Impact of Different Fitness Functions," *2nd Int. Symp. Search Based Softw. Eng.*, no. 25, pp. 89–98, 2010, doi: 10.1109/SSBSE.2010.20.
- [43] J. Murillo-Morera, C. Quesada-López, C. Castro-Herrera, and M. Jenkins, *A genetic algorithm based framework for software effort prediction*, vol. 5, no. 1. Journal of Software Engineering Research and Development, 2017.
- [44] T. R. Benala and R. Mall, "DABE: Differential evolution in analogy-based software development effort estimation," *Swarm Evol. Comput.*, vol. 38, no. July 2017, pp. 158–172, 2018, doi: 10.1016/j.swevo.2017.07.009.
- [45] A. Kaushik, S. Verma, H. J. Singh, and G. Chhabra, "Software cost optimization integrating fuzzy system and COA-Cuckoo optimization algorithm," *Int. J. Syst. Assur. Eng. Manag.*, vol. 8, pp. 1461–1471, 2017, doi: 10.1007/s13198-017-0615-7.
- [46] N. Ghatasheh, H. Faris, I. Aljarah, and R. M. H. Al-Sayyed, "Optimizing Software Effort Estimation Models Using Firefly Algorithm," *J. Softw. Eng. Appl.*, vol. 08, no. 03, pp. 133–142, 2015, doi: 10.4236/jsea.2015.83014.
- [47] V. Resmi, S. Vijayalakshmi, and R. S. Chandrabose, "An effective software project effort estimation system using optimal firefly algorithm," *Cluster Comput.*, vol. 22, no. S5, pp. 11329–11338, Sep. 2019, doi: 10.1007/s10586-017-1388-0.
- [48] F. Marini and B. Walczak, "Particle swarm optimization (PSO). A tutorial," *Chemom. Intell. Lab. Syst.*, vol. 149, pp. 153–165, 2015, doi: 10.1016/j.chemolab.2015.08.020.
- [49] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, 1995, vol. 4, no. 2, pp. 1942–1948, doi: 10.1109/ICNN.1995.488968.
- [50] T. R. Benala and R. Mall, "DABE: Differential evolution in analogy-based software development effort estimation," *Swarm Evol. Comput.*, vol. 38, no. May 2016, pp. 158–172, 2018, doi: 10.1016/j.swevo.2017.07.009.
- [51] Y. Peng, X. Peng, and Z. Liu, "Statistic Analysis on Parameter Efficiency of Particle Swarm Optimization," *Acta Electron. Sin.*, vol. 32, no. 2, pp. 209–2013, 2004.
- [52] A. Agrawal and S. Tripathi, "Particle swarm optimization with probabilistic inertia weight," in *Advances in Intelligent Systems and Computing*, vol. 741, N. Yadav, A. Yadav, J. C. Bansal, K. Deep, and J. H. Kim, Eds. Gurgaon, India: Springer Singapore, 2019, pp. 239–248.
- [53] A. Ratnaweera, S. K. Halgamuge, and H. C. Watson, "Self-Organizing Hierarchical Particle Swarm Optimizer With Time-Varying Acceleration Coefficients," *IEEE Trans. Evol. Comput.*, vol. 8, no. 3, pp. 240–255, Jun. 2004, doi: 10.1109/TEVC.2004.826071.
- [54] A. A. Nagra, H. Fei, Q. H. Ling, and S. Mehta, "An Improved Hybrid Method Combining Gravitational Search Algorithm With Dynamic Multi Swarm Particle Swarm Optimization," *IEEE Access*, vol. 7, no. 2019, 2019, doi: 10.1109/ACCESS.2019.2903137.
- [55] A. Lin, W. Sun, H. Yu, G. Wu, and H. Tang, "Global genetic learning particle swarm

- optimization with diversity enhancement by ring topology,” *Swarm Evol. Comput.*, vol. 44, no. 2019, pp. 571–583, Feb. 2019, doi: 10.1016/j.swevo.2018.07.002.
- [56] R. Vafashoar and M. R. Meybodi, “Cellular learning automata based bare bones PSO with maximum likelihood rotated mutations,” *Swarm Evol. Comput.*, vol. 44, no. 2019, pp. 680–694, 2019.
 - [57] Y. Wang, H. Liu, Z. Yu, and L. Tu, “An improved artificial neural network based on human-behaviour particle swarm optimization and cellular automata,” *Expert Syst. Appl.*, vol. 140, no. 2020, 2020.
 - [58] J. Zhang, J. Sheng, J. Lu, and L. Shen, “UCPSO: A Uniform Initialized Particle Swarm Optimization Algorithm with Cosine Inertia Weight,” *Comput. Intell. Neurosci.*, vol. 2021, pp. 1–18, Mar. 2021, doi: 10.1155/2021/8819333.
 - [59] A. Tharwat, M. Elhoseny, A. E. Hassanien, T. Gabel, and A. Kumar, “Intelligent Bézier curve-based path planning model using Chaotic Particle Swarm Optimization algorithm,” *Cluster Comput.*, vol. 22, no. S2, pp. 4745–4766, Mar. 2019, doi: 10.1007/s10586-018-2360-3.
 - [60] G. Xu *et al.*, “Particle swarm optimization based on dimensional learning strategy,” *Swarm Evol. Comput.*, vol. 45, no. June 2018, pp. 33–51, Mar. 2019, doi: 10.1016/j.swevo.2018.12.009.
 - [61] D. Tian, X. Zhao, and Z. Shi, “DMPSO: Diversity-Guided Multi-Mutation Particle Swarm Optimizer,” *IEEE Access*, vol. 7, no. 3, pp. 124008–124025, 2019, doi: 10.1109/ACCESS.2019.2938063.
 - [62] D. Tian, “Particle Swarm Optimization with Chaos-based Initialization for Numerical Optimization,” *Intell. Autom. Soft Comput.*, vol. 24, no. 2, pp. 331–342, Jun. 2018, doi: 10.1080/10798587.2017.1293881.
 - [63] Z. Ma, X. Yuan, S. Han, D. Sun, and Y. Ma, “Improved Chaotic Particle Swarm Optimization Algorithm with More Symmetric Distribution for Numerical Function Optimization,” *Symmetry (Basel)*, vol. 11, no. 7, p. 876, Jul. 2019, doi: 10.3390/sym11070876.
 - [64] A. Ardiansyah, R. Ferdiana, and A. E. Permanasari, “MUCPSO: A Modified Chaotic Particle Swarm Optimization with Uniform Initialization for Optimizing Software Effort Estimation,” *Appl. Sci.*, vol. 12, no. 3, p. 1081, Jan. 2022, doi: 10.3390/app12031081.
 - [65] K. Ribu, “Estimating Software Projects with Use Cases,” University of Oslo, 2001.
 - [66] R. Silhavy and P. Silhavy, “Using Actors and Use Cases for Software Size Estimation,” *Electronics*, vol. 10, no. 5, pp. 1–20, 2021, doi: 10.3390/electronics10050592.
 - [67] S. Kusumoto, F. Matukawa, and K. Inoue, “Effort Estimation Tool Based on Use Case Points Method,” in *10th International Symposium on Software Metrics*, 2004, pp. 1–3.
 - [68] P. Mohagheghi, B. Anda, and R. Conradi, “Effort estimation of use cases for incremental large-scale software development,” in *Proceedings. 27th International Conference on Software Engineering, 2005. ICSE 2005.*, 2005, pp. 303–311, doi:

- [69] A. Issa, M. Odeh, and D. Coward, "Software Cost Estimation using Use-Case Models: a Critical Evaluation," in *2006 2nd International Conference on Information & Communication Technologies*, 2006, vol. 2, no. 2766–2771, pp. 2766–2771, doi: 10.1109/ICTTA.2006.1684849.
- [70] M. Braz and S. Vergilio, "Software Effort Estimation Based on Use Cases," in *30th Annual International Computer Software and Applications Conference (COMPSAC'06)*, 2006, pp. 221–228, doi: 10.1109/COMPSAC.2006.77.
- [71] B. Anda, H. Dreiem, D. I. K. Sjoberg, and M. Jorgensen, "Estimating Software Development Effort based on Use Cases – Experiences from Industry," in *The Unified Modeling Language. Modeling Languages, Concepts, and Tools*, C. G. Kobryn, Ed. Springer Berlin Heidelberg, 2001.
- [72] J. Jurkiewicz and J. Nawrocki, "Automated events identification in use cases," *Inf. Softw. Technol.*, vol. 58, pp. 110–122, 2015, doi: 10.1016/j.infsof.2014.09.011.
- [73] M. Ochodek, B. Alchimowicz, J. Jurkiewicz, and J. Nawrocki, "Improving the reliability of transaction identification in use cases," *Inf. Softw. Technol.*, vol. 53, no. 8, pp. 885–897, 2011, doi: 10.1016/j.infsof.2011.02.004.
- [74] M. Ochodek and J. Nawrocki, "Automatic Transactions Identification in Use Cases," in *Balancing Agility and Formalism in Software Engineering*, 2008, pp. 55–68.
- [75] G. Robiolo, C. Badano, and R. Orosco, "Transactions and Paths : two use case based metrics which improve the early effort estimation," in *3rd International Symposium on Empirical Software Engineering and Measurement*, 2009, pp. 2–5.
- [76] G. Robiolo and R. Orosco, "Employing use cases to early estimate effort with simpler metrics," *Innov. Syst. Softw. Eng.*, vol. 4, no. 1, pp. 31–43, Apr. 2008, doi: 10.1007/s11334-007-0043-y.
- [77] M. Ochodek, J. Nawrocki, and K. Kwarciak, "Simplifying effort estimation based on Use Case Points," *Inf. Softw. Technol.*, vol. 53, no. 3, pp. 200–213, Mar. 2011, doi: 10.1016/j.infsof.2010.10.005.
- [78] R. Silhavy, P. Silhavy, and Z. Prokopova, "Algorithmic Optimisation Method for Improving Use Case Points Estimation," *PLoS One*, vol. 10, no. 11, p. e0141887, Nov. 2015, doi: 10.1371/journal.pone.0141887.
- [79] H. T. Hoc, V. Van Hai, and H. Le Thi Kim Nhung, "AdamOptimizer for the Optimisation of Use Case Points Estimation," in *Advances in Intelligent Systems and Computing*, vol. 1294, 2020, pp. 747–756.
- [80] R. Silhavy, P. Silhavy, and Z. Prokopova, "Analysis and selection of a regression model for the Use Case Points method using a stepwise approach," *J. Syst. Softw.*, vol. 125, pp. 1–14, Mar. 2017, doi: 10.1016/j.jss.2016.11.029.
- [81] H. Le Thi Kim Nhung, H. T. Hoc, and V. Van Hai, "An Evaluation of Technical and Environmental Complexity Factors for Improving Use Case Points Estimation," in

- [82] H. L. T. K. Nhung, V. Van Hai, R. Silhavy, Z. Prokopova, and P. Silhavy, “Parametric Software Effort Estimation Based on Optimizing Correction Factors and Multiple Linear Regression,” *IEEE Access*, vol. 10, pp. 2963–2986, 2022, doi: 10.1109/ACCESS.2021.3139183.
- [83] K. Qi and B. W. Boehm, “Detailed use case points (DUCPs),” in *Proceedings of the 10th International Workshop on Modelling in Software Engineering - MiSE '18*, 2018, pp. 17–24, doi: 10.1145/3193954.3193955.
- [84] K. Qi, A. Hira, E. Venson, and B. W. Boehm, “Calibrating use case points using bayesian analysis,” in *Proceedings of the 12th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement - ESEM '18*, 2018, pp. 1–10, doi: 10.1145/3239235.3239236.
- [85] K. Rak, Ž. Car, and I. Lovrek, “Effort estimation model for software development projects based on use case reuse,” *J. Softw. Evol. Process*, vol. 31, no. 2, p. e2119, Feb. 2019, doi: 10.1002/smr.2119.
- [86] A. B. Nassif, D. Ho, and L. F. Capretz, “Towards an early software estimation using log-linear regression and a multilayer perceptron model,” *J. Syst. Softw.*, vol. 86, no. 1, pp. 144–160, Jan. 2013, doi: 10.1016/j.jss.2012.07.050.
- [87] Sholiq, T. Sutanto, A. P. Widodo, and W. Kurniawan, “Effort Rate on Use Case Point Method for Effort Estimation of Website Development,” *J. Theor. Appl. Inf. Technol.*, vol. 63, no. 1, pp. 209–218, 2014.
- [88] Subriadi, A. Pribadi, and P. A. Ningrum, “Critical Review of the Effort Rate Value in Use Case Point Method for Estimating Software Development Effort,” vol. 59, no. 3, pp. 735–744, 2014.
- [89] M. Azzeh and A. B. Nassif, “Project productivity evaluation in early software effort estimation,” *J. Softw. Evol. Process*, vol. 30, no. 12, pp. 1–12, 2018, doi: 10.1002/smr.2110.
- [90] M. Azzeh, A. B. Nassif, S. Banitaan, and C. Lopez-Martin, “Ensemble of Learning Project Productivity in Software Effort Based on Use Case Points,” in *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, Dec. 2018, pp. 1427–1431, doi: 10.1109/ICMLA.2018.00232.
- [91] M. Azzeh and A. B. Nassif, “A hybrid model for estimating software project effort from Use Case Points,” *Appl. Soft Comput. J.*, vol. 49, pp. 981–989, 2016, doi: 10.1016/j.asoc.2016.05.008.
- [92] M. Azzeh, A. Bou, N. Cuauhtémoc, and L. Martín, “Empirical analysis on productivity prediction and locality for use case points method,” *Softw. Qual. J.*, no. 0123456789, 2021, doi: 10.1007/s11219-021-09547-0.
- [93] S. K. Rath, B. P. Acharya, and S. M. Satapathy, “Early stage software effort estimation using random forest technique based on use case points,” *IET Softw.*, vol. 10, no. 1, pp. 10–17, 2016, doi: 10.1049/iet-sen.2014.0122.

- [94] Z. Prokopova, R. Silhavy, and P. Silhavy, "The Effects of Clustering to Software Size Estimation for the Use Case Points Methods," in *Software Engineering Trends and Techniques in Intelligent Systems*, 2017, pp. 479–490.
- [95] R. Silhavy, P. Silhavy, and Z. Prokopova, "Evaluating subset selection methods for use case points estimation," *Inf. Softw. Technol.*, vol. 97, no. June 2017, pp. 1–9, 2018, doi: 10.1016/j.infsof.2017.12.009.
- [96] Ardiansyah, R. Ferdiana, and A. E. Permanasari, "Use Case Points based software effort prediction using regression analysis," in *2019 International Conference on Advanced Computer Science and information Systems (ICACSIS)*, Oct. 2019, pp. 15–20, doi: 10.1109/ICACSIS47736.2019.8979851.
- [97] M. R. Singh and S. S. Mahapatra, "A quantum behaved particle swarm optimization for flexible job shop scheduling," *Comput. Ind. Eng.*, vol. 93, pp. 36–44, Mar. 2016, doi: 10.1016/j.cie.2015.12.004.
- [98] H. Zhang, J. Xie, Q. Hu, L. Shao, and T. Chen, "A hybrid DPSO with Levy flight for scheduling MIMO radar tasks," *Appl. Soft Comput. J.*, vol. 71, pp. 242–254, 2018, doi: 10.1016/j.asoc.2018.06.028.
- [99] X. Xu, H. Rong, M. Trovati, M. Liptrott, and N. Bessis, "CS-PSO: chaotic particle swarm optimization algorithm for solving combinatorial optimization problems," *Soft Comput.*, vol. 22, no. 3, pp. 783–795, Feb. 2018, doi: 10.1007/s00500-016-2383-8.
- [100] H. Abdullah, "An Improvement in LQR Controller Design based on Modified Chaotic Particle Swarm Optimization and Model Order Reduction," *Int. J. Intell. Eng. Syst.*, vol. 14, no. 1, pp. 157–168, Feb. 2021, doi: 10.22266/ijies2021.0228.16.
- [101] S. Rahnamayan, H. R. Tizhoosh, and M. M. A. Salama, "Opposition-Based Differential Evolution Algorithms," in *2006 IEEE International Conference on Evolutionary Computation*, 2006, pp. 2010–2017, doi: 10.1109/CEC.2006.1688554.
- [102] H. Wang, Z. Wu, and S. Rahnamayan, "Enhanced opposition-based differential evolution for solving high-dimensional continuous optimization problems," *Soft Comput.*, vol. 15, no. 11, pp. 2127–2140, Nov. 2011, doi: 10.1007/s00500-010-0642-7.
- [103] A. U. Rehman, A. Islam, and S. B. Belhaouari, "Multi-Cluster Jumping Particle Swarm Optimization for Fast Convergence," *IEEE Access*, vol. 8, no. 1, pp. 189382–189394, 2020, doi: 10.1109/ACCESS.2020.3031003.
- [104] H. T. Rauf, U. Shoaib, M. I. Lali, M. Alhaisoni, M. N. Irfan, and M. A. Khan, "Particle Swarm Optimization With Probability Sequence for Global Optimization," *IEEE Access*, vol. 8, pp. 110535–110549, 2020, doi: 10.1109/ACCESS.2020.3002725.
- [105] M. Arif, J. Chen, G. Wang, and H. T. Rauf, "Cognitive population initialization for swarm intelligence and evolutionary computing," *J. Ambient Intell. Humaniz. Comput.*, May 2021, doi: 10.1007/s12652-021-03271-0.
- [106] R. . Eberhart and Y. Shi, "Tracking and optimizing dynamic systems with particle swarms," in *Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No.01TH8546)*, 2001, vol. 1, pp. 94–100, doi: 10.1109/CEC.2001.934376.

- [107] R. Cheng and Y. Jin, "A social learning particle swarm optimization algorithm for scalable optimization," *Inf. Sci. (Ny)*, vol. 291, no. C, pp. 43–60, Jan. 2015, doi: 10.1016/j.ins.2014.08.039.
- [108] X. Zhang, X. Wang, Q. Kang, and J. Cheng, "Differential mutation and novel social learning particle swarm optimization algorithm," *Inf. Sci. (Ny)*, vol. 480, pp. 109–129, Apr. 2019, doi: 10.1016/j.ins.2018.12.030.
- [109] D. Sedighizadeh, E. Masehian, M. Sedighizadeh, and H. Akbaripour, "GEPSo: A new generalized particle swarm optimization algorithm," *Math. Comput. Simul.*, vol. 179, pp. 194–212, Jan. 2021, doi: 10.1016/j.matcom.2020.08.013.
- [110] H. Chen, W. Wang, X. Chen, and L. Qiu, "Multi-objective reservoir operation using particle swarm optimization with adaptive random inertia weights," *Water Sci. Eng.*, vol. 13, no. 2, pp. 136–144, Jun. 2020, doi: 10.1016/j.wse.2020.06.005.
- [111] A. A. Nagra, F. Han, and Q. H. Ling, "An improved hybrid self-inertia weight adaptive particle swarm optimization algorithm with local search," *Eng. Optim.*, vol. 51, no. 7, pp. 1115–1132, Jul. 2019, doi: 10.1080/0305215X.2018.1525709.
- [112] Y. Shi and R. C. Eberhart, "Fuzzy adaptive particle swarm optimization," in *Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No.01TH8546)*, 2001, vol. 1, pp. 101–106, doi: 10.1109/CEC.2001.934377.
- [113] K. Chen, F. Zhou, and A. Liu, "Chaotic dynamic weight particle swarm optimization for numerical function optimization," *Knowledge-Based Syst.*, vol. 139, pp. 23–40, Jan. 2018, doi: 10.1016/j.knosys.2017.10.011.
- [114] H. Koyuncu, "GM-CPSO: A New Viewpoint to Chaotic Particle Swarm Optimization via Gauss Map," *Neural Process. Lett.*, vol. 52, no. 1, pp. 241–266, Aug. 2020, doi: 10.1007/s11063-020-10247-2.
- [115] H. Liu, X. W. Zhang, and L. P. Tu, "A modified particle swarm optimization using adaptive strategy," *Expert Syst. Appl.*, vol. 152, no. 15 August 2020, p. 113353, 2020, doi: 10.1016/j.eswa.2020.113353.
- [116] W. Ye, W. Feng, and S. Fan, "A novel multi-swarm particle swarm optimization with dynamic learning strategy," *Appl. Soft Comput.*, vol. 61, pp. 832–843, Dec. 2017, doi: 10.1016/j.asoc.2017.08.051.
- [117] F. Wang, H. Zhang, K. Li, Z. Lin, J. Yang, and X.-L. Shen, "A hybrid particle swarm optimization algorithm using adaptive learning strategy," *Inf. Sci. (Ny)*, vol. 436–437, no. April 2018, pp. 162–177, Apr. 2018, doi: 10.1016/j.ins.2018.01.027.
- [118] A. Lin, W. Sun, H. Yu, G. Wu, and H. Tang, "Adaptive comprehensive learning particle swarm optimization with cooperative archive," *Appl. Soft Comput.*, vol. 77, no. April 2019, pp. 533–546, Apr. 2019, doi: 10.1016/j.asoc.2019.01.047.
- [119] S. Zhou, L. Sha, S. Zhu, and L. Wang, "Adaptive hierarchical update particle swarm optimization algorithm with a multi-choice comprehensive learning strategy," *Appl. Intell.*, May 2021, doi: 10.1007/s10489-021-02413-3.

- [120] Y. Shi and R. Eberhart, "A modified particle swarm optimizer," in *1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98TH8360)*, 1998, vol. 49, no. 5, pp. 69–73, doi: 10.1109/ICEC.1998.699146.
- [121] H. Maaranen, K. Miettinen, and A. Penttinen, "On initial populations of a genetic algorithm for continuous optimization problems," *J. Glob. Optim.*, vol. 37, no. 3, pp. 405–436, Jan. 2007, doi: 10.1007/s10898-006-9056-6.
- [122] PMI, *A guide to the project management body of knowledge (PMBOK Guide)*, 5th ed. Newton Square, PA: Project Management Institute, 2013.
- [123] A. Trendowicz and R. Jeffery, *Software Project Effort Estimation: Foundation and Best Practice Guidelines for Success*. Springer, 2014.
- [124] K. E. Wiegers and J. Beatty, *Software Requirements*, 3rd ed. Washington: Microsoft Press, 2013.
- [125] I. Jacobson, "Object-oriented development in an industrial environment," in *Conference proceedings on Object-oriented programming systems, languages and applications - OOPSLA '87*, 1987, pp. 183–191, doi: 10.1145/38765.38824.
- [126] I. Jacobson, *Object-Oriented Software Engineering: A Use Case Driven Approach*. Addison Wesley, 1992.
- [127] Sholiq, A. P. Subriadi, F. A. Muqtadiroh, and R. S. Dewi, "A model of owner estimate cost for software development project in Indonesia," *J. Softw. Evol. Process*, vol. 31, no. 10, 2019, doi: 10.1002/smr.2175.
- [128] J. Xin, G. Chen, and Y. Hai, "A particle swarm optimizer with multi-stage linearly-decreasing inertia weight," *Proc. 2009 Int. Jt. Conf. Comput. Sci. Optim. CSO 2009*, vol. 1, no. February, pp. 505–508, 2009, doi: 10.1109/CSO.2009.420.
- [129] Y. Shi and R. C. Eberhart, "Empirical study of particle swarm optimization," *Proc. 1999 Congr. Evol. Comput. CEC 1999*, vol. 3, pp. 1945–1950, 1999, doi: 10.1109/CEC.1999.785511.
- [130] T. Menzies, "Nasa93 [Data set]." Zenodo, 2008, doi: <http://doi.org/10.5281/zenodo.268419>.
- [131] Z. Zia, S. K. Tipu, and S. Zia, "An Effort Estimation Taxonomy for Agile Software Development," *Adv. Comput. Sci. its Appl.*, vol. 2, no. 1, pp. 314–324, May 2012, [Online]. Available: <https://www.worldscientific.com/doi/abs/10.1142/S0218194017500243>.
- [132] R. Silhavy, "Use Case Points Benchmark Dataset v1." Mendeley Data, 2017, doi: 10.17632/2rfkjh3cn.1.
- [133] A. Hira, S. Sharma, and B. Boehm, "Calibrating COCOMO® II for projects with high personnel turnover," *Proc. Int. Work. Softw. Syst. Process - ICSSP '16*, pp. 51–55, 2016, doi: 10.1145/2904354.2904367.
- [134] T. T. Khuat and M. H. Le, "A Novel Hybrid ABC-PSO Algorithm for Effort Estimation of

Software Projects Using Agile Methodologies,” *J. Intell. Syst.*, vol. 27, no. 3, pp. 489–506, 2018, doi: 10.1515/jisys-2016-0294.

- [135] R. E. (Dick) Fairley R.E., “The influence of COCOMO on software engineering education and training,” *J. Syst. Softw.*, vol. 80, no. 8, pp. 1201–1208, 2007, doi: 10.1016/j.jss.2006.09.044.
- [136] E. Alpaydin, *Introduction to machine learning*. MIT press, 2014.
- [137] E. Kocaguneli and T. Menzies, “Software effort models should be assessed via leave-one-out validation,” *J. Syst. Softw.*, vol. 86, no. 7, pp. 1879–1890, 2013, doi: 10.1016/j.jss.2013.02.053.
- [138] Q. Li, Q. Wang, Y. Yang, and M. Li, “Reducing biases in individual software effort estimations,” in *Proceedings of the Second ACM-IEEE international symposium on Empirical software engineering and measurement - ESEM '08*, 2008, p. 223, doi: 10.1145/1414004.1414041.
- [139] M. Shepperd and S. MacDonell, “Evaluating prediction systems in software project estimation,” *Inf. Softw. Technol.*, vol. 54, no. 8, pp. 820–827, Aug. 2012, doi: 10.1016/j.infsof.2011.12.008.
- [140] P. D. Ellis, *The essential guide to effect sizes: statistical power, meta analysis, and the interpretation of research results*. Cambridge: Cambridge University Press, 2010.
- [141] J. Cohen, “A power primer,” *Psychol. Bull.*, vol. 112, no. 1, pp. 155–159, 1992, doi: 10.1037/0033-2909.112.1.155.
- [142] A. P. Piotrowski, J. J. Napiorkowski, and A. E. Piotrowska, “Population size in Particle Swarm Optimization,” *Swarm Evol. Comput.*, vol. 58, no. May, p. 100718, 2020, doi: 10.1016/j.swevo.2020.100718.
- [143] I. D. Kenestie and Sholih, “Determining Effort Rate (ER) Value for Use Case Points based Educational Software Development Effort Estimation,” *J. Tek. POMITS*, pp. 1–11, 2011.
- [144] M. Harman, “The current state and future of search based software engineering,” in *FoSE 2007: Future of Software Engineering*, 2007, pp. 342–357, doi: 10.1109/FOSE.2007.29.
- [145] B. Boehm, B. Clark, E. Horowitz, C. Westland, R. Madachy, and R. Selby, “Cost models for future software life cycle processes: COCOMO 2.0,” *Ann. Softw. Eng.*, vol. 1, no. 1, pp. 57–94, Dec. 1995, doi: 10.1007/BF02249046.
- [146] S. Baluja and R. Caruana, “Removing the Genetics from the Standard Genetic Algorithm,” in *Machine Learning Proceedings 1995*, Elsevier, 1995, pp. 38–46.
- [147] M. Azzeh, A. B. Nassif, S. Banitaan, and F. Almasalha, “Pareto efficient multi-objective optimization for local tuning of analogy-based estimation,” *Neural Comput. Appl.*, vol. 27, no. 8, pp. 2241–2265, Nov. 2016, doi: 10.1007/s00521-015-2004-y.
- [148] V. Khatibi Bardsiri, D. N. A. Jawawi, S. Z. M. Hashim, and E. Khatibi, “A PSO-based model to increase the accuracy of software development effort estimation,” *Softw. Qual. J.*, vol. 21, no. 3, pp. 501–526, 2013, doi: 10.1007/s11219-012-9183-x.



UNIVERSITAS
GADJAH MADA

MUCPSO: Metode Estimasi Usaha Perangkat Lunak Menggunakan Uniform Chaotic Personal Particle Swarm Optimization

ARDIANSYAH, Dr. Ir Ridi Ferdiana, S.T., M.T., IPM; Adhistya Erna Permanasari, S.T., M.T., Ph.D

Universitas Gadjah Mada, 2022 | Diunduh dari <http://etd.repository.ugm.ac.id/>

- [149] W. B. Langdon, J. Dolado, F. Sarro, and M. Harman, “Exact Mean Absolute Error of Baseline Predictor, MARP0,” *Inf. Softw. Technol.*, vol. 73, pp. 16–18, May 2016, doi: 10.1016/j.infsof.2016.01.003.
- [150] T. Bartz-Beielstein and M. Preuss, “Experimental research in evolutionary computation,” in *Proceedings of the 2007 GECCO conference companion on Genetic and evolutionary computation - GECCO '07*, 2007, p. 3001, doi: 10.1145/1274000.1274102.